

Herramientas de soporte para temas de Comunicación de Datos

Guillermo Rigotti

UNICEN – Fac. de Ciencias Exactas

ISISTAN – Grupo de Objetos y Visualización

Pje. Arroyo Seco, (7000) Tandil, Bs. As. Argentina

TE: +54-2293-440363 FAX: +54-2293-440362

Email: grigotti@exa.unicen.edu.ar

Resumen

Como consecuencia de las dificultades experimentadas por los alumnos para asimilar algunos de los conceptos de las materias del área Comunicación de Datos, surge la idea de desarrollar un soporte que facilite la comprensión de los mismos a través de la programación y/o visualización de las interacciones entre las entidades que se comunican entre sí en cada caso. El soporte se compone de diferentes módulos independientes entre sí pero homogéneos tanto en el aspecto referente a las herramientas utilizadas para su desarrollo como al referido a la modalidad de uso por parte de los alumnos. Cada uno de estos módulos trata un tema en particular y permite la realización de trabajos prácticos relativos al mismo.

En el presente trabajo se presenta la estructura general de dichos módulos y se describe brevemente aquellos ya desarrollados y los que actualmente se están desarrollando.

1. Introducción

El presente trabajo surge como consecuencia de observar las dificultades experimentadas por los alumnos para la asimilación de conceptos básicos de comunicación de datos, sobre todo en sus pasos iniciales en el área.

El objetivo perseguido es el desarrollo de un ambiente homogéneo de rápida asimilación por parte de los alumnos, que posibilite la comprensión y el manejo de los conceptos de mayor importancia a través de la programación y/o visualización de interacciones entre los procesos que componen los protocolos. Este ambiente, integrado por diferentes módulos, debe ser lo suficientemente flexible para que puedan plasmarse conceptos de diferente complejidad, desde aquellos abstractos y relativamente simples de las arquitecturas de niveles hasta los más complejos y dinámicos que representan interacciones entre procesos remotos asincrónicos comunicándose a través de medios que introducen errores y demoras, en ciertos casos variables.

2. Características de los diferentes módulos

Con el objeto de que el uso de los distintos módulos demande un mínimo esfuerzo por parte de los alumnos, en todos ellos se conserva la misma estructura y se utiliza el mismo lenguaje de programación para especificar las interacciones. De esta manera, resultan las siguientes características comunes a todos ellos:

- Rapidez de aprendizaje del lenguaje a utilizar por parte de los alumnos
- Sencillez del soporte provisto
- Portabilidad de los programas de simulación y de visualización.

La estructura común de cada módulo se muestra en la figura 1.

El usuario debe describir las interacciones a simular en lenguaje Tcl [Osterhout,1994]. Dicha descripción es dependiente de cada caso en particular, pudiendo tener diferente complejidad. Por ejemplo, la especificación de interacciones de interfaz entre dos niveles ISO es simple, consistiendo solo de la indicación de la primitiva a ejecutar entre un par de niveles, mientras que la especificación de un protocolo de nivel 2 implica la programación del mismo.

La especificación realizada por los alumnos es interpretada por el soporte de programación provisto y simulada utilizando el scheduler provisto por Ns [Fall,2000]. Dependiendo de cada caso en particular, dicho soporte puede involucrar la totalidad de la funcionalidad relativa al problema que se está tratando¹ o parte de ella². El resultado de este proceso de simulación consiste de dos archivos.

Uno de ellos contiene una descripción textual de las diferentes acciones de las entidades involucradas, y su objetivo es que sea utilizado para comprobar el comportamiento de las mismas. Dependiendo del problema en particular, es posible procesar posteriormente esta salida para obtener información estadística³

El segundo archivo contiene los eventos relevantes para la visualización. Por cada evento se indica el tiempo de comienzo, el tiempo de finalización y los parámetros correspondientes al tipo de evento, que se utilizarán en la visualización. Esta manera de especificar los eventos permite mayor flexibilidad en el momento de visualizar las interacciones, ya que es posible adelantar o atrasar el tiempo corriente y definir el paso de simulación con el detalle deseado soportado por el programa de simulación.

3. Módulos de interés

Los módulos ya desarrollados y en desarrollo responden a intereses específicos de las cátedras del área. La información para el desarrollo de los mismos proviene fundamentalmente de bibliografía de las materias involucradas [Tanenbaum,1996] [Halsall,1992] entre otros, a la cual se le agregan normas específicas en cada caso. Algunos de estos módulos son desarrollados por la cátedra mientras que otros son desarrollados por alumnos en calidad de trabajos especiales con supervisión de la cátedra.

3.1. Módulos desarrollados

¹ Por ejemplo en el caso de packet switching, dicho módulo implementa todas las funciones de transporte de paquetes a través de la red, asignación de buffers, etc

² En el caso de un soporte para la programación de protocolos de nivel 2, se incluirá la funcionalidad correspondiente al soporte ofrecido por el equipo (hardware más sistema operativo) y al del link.

³ Por ejemplo, obtener la eficiencia de un protocolo de nivel dos para diferentes configuraciones del tamaño de ventana.

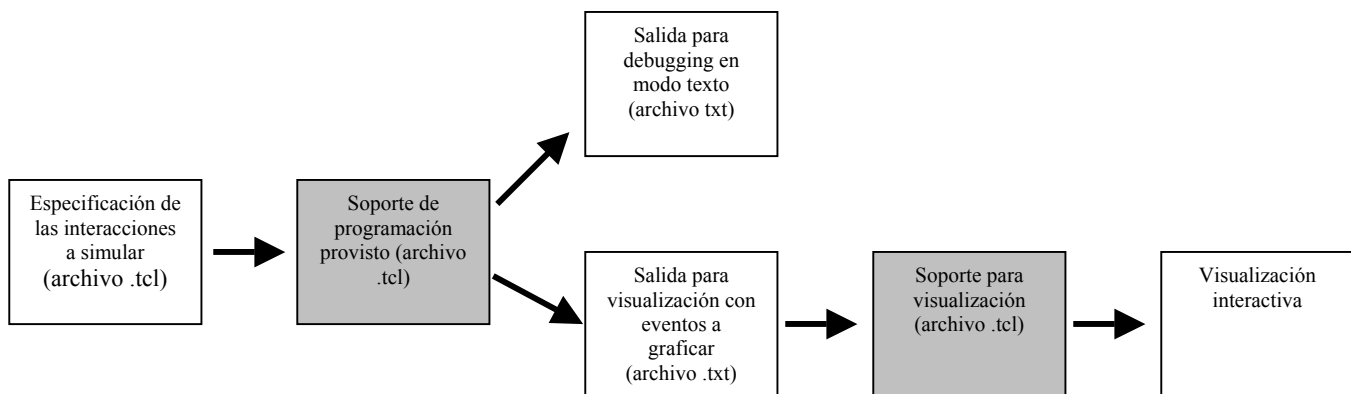


Fig. 1. Módulos que componen el sistema desarrollado (sombreados)

3.1.1 Programación de protocolos a nivel 2

El objetivo de este trabajo es doble. En primer término posibilita que los alumnos puedan plasmar a través de una programación muy simple algunos de los conceptos presentes en las arquitecturas de niveles descritas en el modelo OSI/ISO, tales como primitivas de interfaz, puntos de acceso al servicio (SAPs), etc. En una etapa posterior el soporte es utilizado para el desarrollo de protocolos simples de nivel 2. En esta etapa se pretende que los alumnos se familiaricen con la programación del software de comunicaciones (manejo de buffers, etc) y con la programación de las interacciones que tienen lugar entre procesos que corren de manera asincrónica y se comunican a través de un medio que introduce demoras y posibles errores. Trabajo desarrollado por la cátedra..

3.1.2 Relación entre las secuencias de primitivas en niveles OSI/ISO

El objetivo es que los alumnos comprendan la relación entre las acciones de los diferentes niveles de una arquitectura. Es posible especificar varios niveles adyacentes, cada uno de ellos con sus características propias (orientado o no a conexión, confirmado, etc). Los alumnos pueden especificar las primitivas a ser ejecutadas por un nivel (connect.request, data.request, etc) y observar el efecto que estas producen en el equipo local, en la línea de comunicaciones y en el equipo remoto. Trabajo desarrollado por Ignacio Marcovecchio.

3.1.3 Programación de interacciones en DQDB

El objetivo es que los alumnos puedan realizar un seguimiento del protocolo DQDB. Es posible especificar los parámetros de operación del protocolo y también los requerimientos de transmisión de las diferentes estaciones, en base a los cuales se observará de qué manera las mismas reservan canal y como opera la cola distribuida de mensajes a transmitir en cada uno de los sentidos. Es posible especificar el uso de prioridades. Trabajo desarrollado por Pablo Vagliati.

3.1.4 Programación y visualización del protocolo CSMA/CD

El objetivo es que los alumnos puedan observar en detalle el comportamiento del protocolo CSMA/CD a través de la programación simple del mismo (especificación de bloques a enviar por parte de cada equipo) y la visualización de los bloques y colisiones en el canal. Trabajo desarrollado por Cristina Rodríguez.

3.2 Módulos en desarrollo

3.2.1 Soporte para la programación de protocolos de acceso al medio

Permite el desarrollo de protocolos de acceso al medio operando sobre vínculos multiacceso broadcast. Es posible especificar parámetros del canal, tales como velocidad de transmisión y demora de propagación y tipo de señales soportadas (datos, ocioso, colisión). El objetivo es que los alumnos programen los métodos de control de acceso al medio clásicos (aloha, CSMA/CD, basic bit map, etc) y comprueben su funcionamiento a través de una salida en modo texto. Es posible además obtener estadísticas relativas al comportamiento de los diferentes protocolos a través del proceso del archivo de salida. Un posible agregado lo constituye un módulo de visualización.

3.2.2 Programación de diferentes mecanismos de ventana en el nivel 2

El objetivo es que los alumnos puedan configurar mecanismos de control de flujo a nivel 2, indicando parámetros como la cantidad de buffers en el emisor y el receptor, velocidad de transmisión, tiempos de espera por distintos eventos (piggybacking, retransmisión), números máximos de secuencia de paquetes, etc. Este módulo apunta al trabajo con el mecanismo de ventana, haciéndolo independiente de otros aspectos del protocolo. A través de la salida en modo texto y en modo gráfico podrán observar el comportamiento de las entidades emisora y receptora en condiciones normales y bajo condiciones de error tanto de especificación de parámetros (por ejemplo tiempos de espera muy cortos) como de transmisión (bloques erróneos o perdidos). Este trabajo se desarrolla teniendo como base el soporte de programación de protocolos a nivel 2.

3.2.3 Especificación de tráfico y visualización para redes de packet switching

Permite configurar una red de packet switching limitada a 9 nodos. Es posible especificar el tráfico en la red generado por cada nodo (tiempos de envío, volumen y tamaño de los paquetes), la capacidad de los vínculos que conectan los nodos y la cantidad de buffers. El objetivo es mostrar gráficamente la operación de la red, resaltando la ocupación de buffers, el pipelining en transmisión y el posible descarte de paquetes. La salida en modo texto está estructurada de manera que sea posible extraer información estadística (paquetes perdidos, etc) a través de un proceso simple con utilitarios de procesamiento de texto como awk.

3.2.4 Programación y visualización del funcionamiento de una UART

Permite configurar los parámetros de operación de una UART simple con el objeto de observar el intercambio de caracteres entre los equipos, el manejo de buffers y los posibles errores producidos por problemas de configuración.

3.2.5 Soporte para la programación, visualización y evaluación de TCP sobre diferentes tipos de nivel de red.

El objetivo es proveer un soporte de red emulado y parametrizable que produzca condiciones de operación similares a las reales para el protocolo TCP. Los alumnos podrán programar características de TCP (slow start, etc.) y comprobar de qué manera se comporta el protocolo en diferentes situaciones.

4. Conclusiones

En lo que respecta al desarrollo de los diferentes módulos, se ha logrado un grado de homogeneidad que permite su rápida asimilación y mantenimiento. Las experiencias realizadas en las clases prácticas fueron satisfactorias en lo que respecta al único módulo con el que se ha trabajado hasta el momento, programación de protocolos a nivel 2. Los demás módulos ya implementados se utilizarán en las cátedras correspondientes durante el año 2003.

5. Bibliografia

[Fall, 2000] K. Fall (ed), “Ns Notes and Documentation” VINT Project, UC Berkeley, LBL, USC/ISI, Xerox PARC, March 2000.

[Halsall, 1992], “Data Communications, Computer Networks and Open Systems”, Halsall, F., Addison-Wesley, 1992.

[Osterhout,1994].

[Tanenbaum, 1996] “Computer Networks” 3rd edition, Tanenbaum, A., Prentice Hall, 1996.